Thekla – Visual Semantic Analysis of Document Clusters

Johannes Knopp and Johanna Völker**

Data & Web Science Research Group
University of Mannheim, Germany
{johannes,johanna}@informatik.uni-mannheim.de

Abstract. We present *thekla*, a software that creates semantic visualizations of document clusters. The system uses topic models in order to distinguish between different aspects of the clusters' semantics, and thus facilitates a fine-grained analysis of the thematic similarities and differences between clusters.

Keywords: cluster visualization, topic models, lexical semantics

1 Introduction

Many problems in NLP can be stated as a document clustering problem, for example Information Retrieval, Word Sense Disambiguation, and Text Summarization.[5, 6, 8] However, it is often hard for humans to interpret the resulting clusters or compare them to each other because documents are mostly represented using a vector space model which has the advantage of enabling distance computation between documents, but the drawback of being hard to interpret for humans. To deal with this problem we present *thekla*, a tool to create meaningful visualizations of document clusters. Thekla makes use of topic models to represent the clusters in a meaningful way.

A Topic Model (TM)[7] is a generative model that describes documents as a mixture of (latent) topics which are in return represented by a distribution over words. The topic-document distributions and the topics are inferred from the corpus. The intuition of a topic is that it has a predominant theme and ranks the words with accordance to it which means that a word with a high probability for a topic is relevant to the topic's theme. Following this thought we will represent each word in the vocabulary with the help of topics and hereby create a *topic signature* for it. In more formal terms, the topic signature *tsig* for word w_j is composed of the word's probabilities in the topics $t_1, \ldots, t_i$:

$$tsig(w_j) = \langle p(w_j|t_1), p(w_j|t_2), \ldots, p(w_j|t_i) \rangle \tag{1}$$

A document can be represented by aggregating its words' topic signatures, e.g. by computing their arithmetic mean. Accordingly, the semantics of a cluster

** financed by a Margarete-von-Wrangell scholarship of the European Social Fund (ESF) and the Ministry of Science, Research and the Arts Baden-Württemberg

can be represented by computing the centroid of the documents that belong to one cluster. Topics can be interpreted by humans by listing their most probable words. Thekla visualizes cluster centroids by plotting each cluster in a spider diagram[1] that lists the top topic words at the axes. In the following, we will describe the technical details of thekla and demonstrate some spider diagrams that result vom visualizing a gold standard clustering.

2 Thekla

Thekla is written in Python[2] and makes use of the libraries matplotlib[2][3] and numpy[4]. It can be started via command line and load configuration files containing the nessary information to create and store the visualization of a clustering. The dummy of a configuration file is presented in Figure 1.

```
#comments start with "#"
#configuration sections are marked by brackets
[topicmodel]
#the vocabulary of the topic model
vocabfile = path/to/vocab/file.vocab
#the output of ldac
betafile = path/to/ldac/result/final.beta

[documents]
#directory of the clustered text documents
docdir = path/to/corpus/

[clustering]
#title used in the visualization
title = cluster_title
#file with the clusters that will be visualized
clusterfile = path/to/result.cluster

[visualization]
res_dir = path/to/save/resulting/visualizations/
#number of best topic words to be displayed
nwords = 7
```

Fig. 1: Example for a configuration file.

```
#one section per cluster
[cluster1]
#list of documents in cluster1
docs: ["path/to/corpus/text_1.txt",
  "path/to/corpus/text_3.txt",
  "path/to/corpus/text_7.txt"]

[cluster2]
#list of documents in cluster2
docs: ["path/to/corpus/text_2.txt",
  "path/to/corpus/text_4.txt",
  "path/to/corpus/text_8.txt"]

[..]
```

Fig. 2: Example for a cluster file.

Thekla assumes that the documents are actually files and that the topic models are created with David Blei's TM implementation *ldac*[5], so we can load the `final.beta` file which is the output of *ldac*. After that all text documents in the provided `docdir` are loaded and represented with their topic signature as described in the introduction. Subsequently, the `clusterfile` is loaded which follows the example in Figure 2. For each cluster the documents' topic signatures

[1] Also known as radar chart.

[2] `http://python.org`

[3] `http://matplotlib.org/index.html`

[4] `http://www.numpy.org/`

[5] Available at `http://www.cs.princeton.edu/~blei/lda-c/index.html`

are used to compute a centroid which is then visalized and saved in *.png* format in the specified `res_dir` with `title` as its name. If no clustering is present thekla also can perform its own clustering based on the topics which is explained in [3]. We plan to make the code available at `https://github.com/joknopp/thekla`.

3 Demonstration

To demonstrate thekla we visualized the gold standard of the Word Sense Induction task of SemEval 2010 [4]. The SemEval data provided a training and a test set for ambiguous words. For each ambiguous word there is a collection of text fragments where the respective word is used in one specific sense. We consider each fragment as a document for the TM creation, but first they were preprocessed and only the nouns were kept.

The number of topics has to be chosen based on the level of detail that is of interest for the task at hand and the size of the corpus. The bigger the corpus size, the more detailed meaningful topics may emerge for a high number of topics. In this case we built several topic models while setting the number of topics to $3 \ldots 10$.

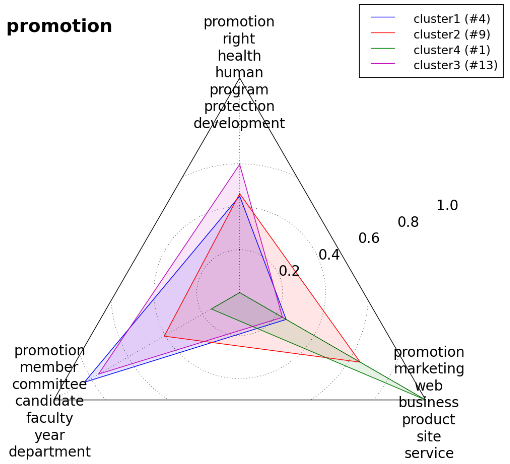

Fig. 3: Visualization of the gold standard clustering of the word "promotion" using 3 topics.

A spider diagram visualization of two of the cluster results for the word "promotion" are presented in Figures 3 and 4. Each dimension corresponds to one

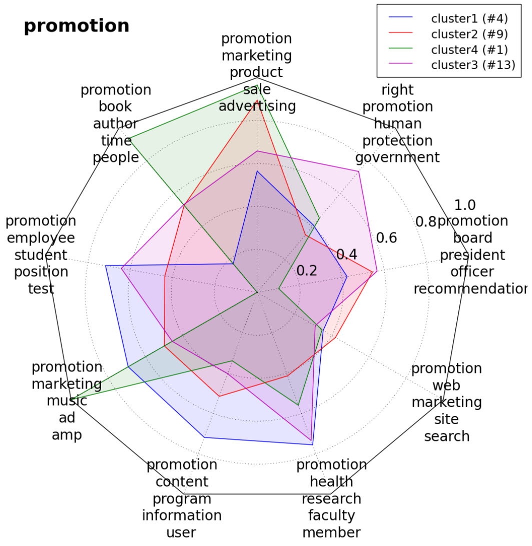

Fig. 4: Visualization of the gold standard clustering of the word "promotion" using 9 topics.

topic and is labeled with the respective most probable words. Each cluster centroid is a vector that spans a plane in the diagram, the number of documents per cluster is specified next to the cluster name in the legend. The word "promotion" is dominant in every topic because it appears in every document. As you can see with a higher number of topics the clusters can be compared in more detail. For example with the number of topics set to 3, there is one topic with the theme "advertising" (lower right corner) while when using 9 topics there are several topics that deal with aspects of advertising.

4 Conclusion

We presented thekla, a Python program that visalualizes document clusters in a semantically interpretable way. This could be helpful for tasks where finding out the semantic differences between clusters is helpful, e.g. near-synonym detection [1] or Word Sense Disambiguation. In the future we plan to enhance thekla to be able to load different topic model formats and investigate alternative aggregation function to create a topic signature from the words in a document.

References

1. Hirst, G.: Near-synonymy and the structure of lexical knowledge. In: AAAI Symposium on Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity. (1995) 51–56

2. Hunter, J.D.: Matplotlib: A 2d graphics environment. Computing In Science & Engineering **9**(3) (2007) 90–95
3. Knopp, J., Völker, J., Ponzetto, S.P.: Topic modeling for word sense induction. In: Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology 2013, Darmstadt, Germany To appear.
4. Manandhar, S., Klapaftis, I., Dligach, D., Pradhan, S.: Semeval-2010 task 14: Word sense induction & disambiguation. In: Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, Association for Computational Linguistics (July 2010) 63–68
5. Navigli, R.: Word sense disambiguation: A survey. ACM Comput. Surv. **41**(2) (February 2009) 10:1–10:69
6. Nomoto, T., Matsumoto, Y.: A new approach to unsupervised text summarization. In: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '01, New York, NY, USA, ACM (2001) 26–34
7. Steyvers, M., Griffiths, T.: Probabilistic topic models. In Landauer, T., Mcnamara, D., Dennis, S., Kintsch, W., eds.: Latent Semantic Analysis: A Road to Meaning. Laurence Erlbaum (2007)
8. Zamir, O., Etzioni, O.: Web document clustering: a feasibility demonstration. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '98, New York, NY, USA, ACM (1998) 46–54